

Санкт–Петербургский государственный университет

БОГДАНОВ Николай Андреевич

Выпускная квалификационная работа

***Разработка и сравнение программ распознавания
вирусной и бактериальной пневмонии на основе
анализа рентгеновских снимков.***

Уровень образования: бакалавриат

Направление 01.03.02 «Прикладная математика и информатика»

Основная образовательная программа СВ.5005.2015 «Прикладная
математика, фундаментальная информатика и программирование»

Профиль «Математическое и программное обеспечение вычислительных
машин»

Научный руководитель:

профессор, кафедра технологии программирования,
д.т.н. Печников Андрей Анатольевич

Рецензент:

научный сотрудник,
Институт прикладных математических исследований КарНЦ РАН,
к.т.н. Никитина Наталия Николаевна

Санкт-Петербург

2021 г.

Содержание

Введение	4
Постановка задачи	5
Обзор литературы	6
Глава 1. Описание медицинских и математических аспектов задачи распознавания пневмонии.	7
1.1. Описание пневмонии.	7
1.2. Диагностика пневмонии.	9
1.3. Описание задачи классификации.	12
1.4. Распознавание пневмонии как задача бинарной классификации.	14
Глава 2. Описание используемых подходов.	16
2.1. 1. Подход, основанный на использовании сверточных нейронных сетей.	16
2.1.1 Функции активации.	17
2.1.2 Функция потерь.	19
2.1.3 Переобучение.	21
2.1.4 Сверточные нейронные сети.	23
2.1.5 Структура нейронной сети, решающей задачу распознавания пневмонии.	27
2.2. Подход, основанный на использовании колмогоровской сложности.	30
2.2.1 Основные теоретические аспекты.	30
2.2.2 Алгоритм классификации.	32
2.2.3 Детали реализации алгоритма классификации.	33
Глава 3. Сравнение результатов.	35
3.1. Описание данных.	35
3.2. Метрики.	36
3.3. Выбор гиперпараметров и сравнение методов.	37
Выводы	41
Заключение	42

Список литературы	43
-----------------------------	----

Введение

Диагностика заболеваний, поражающие органы дыхания, является актуальной медицинской проблемой. Такие болезни чрезвычайно опасны, но своевременное обращение к специалистам обычно помогает избежать тяжелых осложнений. Пневмония является тяжелым инфекционным поражением легочной ткани. Клинические картины вирусной и бактериальной пневмонии во многом совпадают [1].

Наиболее точным методом распознавания пневмонии является компьютерная томография [2], однако компьютерные томографы пока есть далеко не во всех лечебных учреждениях. Основным методом диагностики, использующимся на практике, является рентгенологическое исследование. Данный метод обладает меньшей разрешающей способностью и наличием шумов, но более распространен из-за большей доступности рентгеновских аппаратов [3]. Нагрузка на специалистов-рентгенологов такова, что в течение дня ими анализируются десятки и даже сотни снимков. Это приводит к неизбежным ошибкам при постановке диагнозов. Поэтому актуальна разработка быстрых и надежных компьютерных программ, помогающих врачам оперативно обнаруживать заболевание по рентгеновским снимкам грудной клетки.

В данной работе для решения задачи распознавания по рентгеновским снимкам такого заболевания как пневмония рассматриваются два подхода. Первый из них, активно изучаемый в последнее время, основан на применении нейронных сетей [4-7]. В работах показана перспективность данного подхода. При этом отмечается, что точность классификации сильно зависит от количества представленных образцов, и уделяется особое внимание предварительной обработке изображений.

Второй подход, предлагаемый автором, использует понятие нормализованное расстояние сжатия [8, 9], являющегося практической реализацией такого теоретического понятия как колмогоровская сложность [10-12]. Сама идея классификации изображений с использованием колмогоровской сложности не нова (например, [13]). Однако представляется важным сравнить распространенный сегодня подход, основанный на нейронных сетях,

и подход, базирующийся на колмогоровской сложности, с целью выявления достоинств и недостатков каждого из них, что в дальнейшем будет учитываться при разработке программ для диагностики пневмонии.

В первой главе дается медицинское описание пневмонии и перечисляются основные методы ее диагностирования. Затем приводится способ сведения медицинской задачи обнаружения пневмонии к математической задаче бинарной классификации.

Во второй главе рассматриваются основные теоретические аспекты двух подходов и описываются способы программной реализации каждого из них

Третья глава посвящена сравнению результатов работы алгоритмов на заданном наборе данных. Далее следуют выводы с описанием основных преимуществ и недостатков каждого из подходов.

Постановка задачи

Выпускная квалификационная работа посвящена решению следующих задач:

1. Дать медицинское описание пневмонии и методам ее диагностики.
2. Сформулировать математическую задачу классификации рентгеновских снимков.
3. Описать теоретические основы рассматриваемых подходов к классификации.
4. Разработать программное обеспечение, практически реализующее каждый из алгоритмов.
5. Дать описание исходным данным, протестировать алгоритмы на этих данных.
6. Сравнить производительность алгоритмов, определить их преимущества и недостатки.

Обзор литературы

В [1] содержатся общие сведения о пневмонии, симптомы и способы профилактики. Статья [2] содержит описание особенностей применения метода компьютерной томографии для диагностики пневмонии, когда как [3] содержит информацию о практике применения рентгенографии для тех же целей. В [4-7] приводятся примеры сверточных нейронных сетей, решающих задачу диагностирования пневмонии на основе классификации рентгеновских снимков. В [8] описывается применение алгоритма, использующего колмогоровскую сложность для кластеризации объектов различной природы, таких как генетические последовательности различных животных или отрывки литературных произведений. В [9] описывается такое понятие как нормализованное расстояние сжатия и особенности данной метрики. Идея А.Н. Колмогорова, служащая базой для самого понятия колмогоровской сложности, изложена в [10]. Книга [11] содержит определение колмогоровской сложности в том виде, в котором данное определение используется в ВКР. В [12] доказывается теорема о симметрии алгоритмической информации. Как пример использования колмогоровской сложности и нормализованного расстояния сжатия для классификации изображений можно привести [13]. Набор данных, на котором были протестированы программы, разработанные в рамках ВКР, доступен по ссылке [15], в качестве альтернативного набора данных может быть рассмотрен [14]. Влияние аугментации данных на итоговое качество классификации, производимой сверточной нейронной сетью, рассмотрено в [16]. В [17] проводится сравнения пригодности различных программ-архиваторов для использования в программах, основанных на колмогоровской сложности. Анализ основных функций, применяемых для оценки качества бинарной классификации, приведен в [18].

Глава 1. Описание медицинских и математических аспектов задачи распознавания пневмонии.

1.1 Описание пневмонии.

Пневмонию можно описать как воспалительный процесс в легких, в первую очередь поражающий небольшие воздушные мешки - альвеолы. Альвеолы участвуют в процессе дыхания, осуществляя газообмен с легочными капиллярами. В каждом из легких находится около трехсот миллионов альвеол.

Пневмония является важнейшей отдельно взятой инфекционной причиной детской смертности. В 2017 году от пневмонии умерли 808694 детей в возрасте до 5 лет, это составляет 15 процентов всех случаев смерти детей до 5 лет во всем мире [1].

Пневмония распространена повсеместно, но в наибольшей степени от нее страдают жители развивающихся стран.

Среди возбудителей болезни можно выделить следующие:

1. бактерии:

- стафилококк
- пневмококк
- стрептококк
- гемофильная палочка.

Бактерии являются основной причиной пневмонии, полученной за пределами здравоохранительной системы (передача от человека к человеку). Количество подтипов этих бактерий, устойчивых к антибиотикам, растет с каждым годом.

2. пневмотропные вирусы:

- коронавирус
- грипп
- рс-вирус

- аденовирус.

Вирусы вызывают около трети случаев пневмонии среди взрослых и около 15% случаев среди детей. Эти микроорганизмы могут попасть в легкие различными путями. Одни вирусы попадают в организм когда человек дотрагивается до зараженной поверхности, а затем трогает свои глаза или нос, другие распространяются воздушно-капельным путем.

3. грибки:

- кандида
- аспергилл и т.д

Пневмония, вызванная грибами, достаточно редкое явление и обычно наблюдается у людей с подавленной иммунной системой, например больных СПИДом или принимающих иммунодепрессанты. Количество случаев грибковой пневмонии выросло во второй половине двадцатого века из-за увеличения доли людей с подавленным иммунитетом.

4. паразиты

- *Toxoplasma gondii*
- *Strongyloides stercoralis*
- *Ascaris lumbricoides*
- *Plasmodium malariae*
- *Paragonimus westermani*

Множество паразитов обладает возможностью воздействовать на легкие. Эти организмы обычно попадают в тело через прямой контакт с кожей, проглатывание или укус насекомого. Большинство паразитов, кроме *Paragonimus westermani*, не влияют на легкие в первую очередь, но могут поразить легкие как вторичную цель.

В зависимости от возбудителя, симптомами заболевания могут являться:

- влажный или сухой кашель
- боль в груди
- повышение температуры тела
- затрудненное дыхание
- слабость
- головные боли
- потеря аппетита
- низкое кровяное давление
- повышенный пульс
- боль в суставах

1.2 Диагностика пневмонии.

К основным методам диагностики пневмонии можно отнести следующие:

1. Рентгенография грудной клетки.
2. Компьютерная томография грудной клетки.
3. Микроскопическое исследования мокроты с окраской по Граму.
4. Посев мокроты на питательные среды.
5. Общий и биохимический анализ крови и другие методы.

Два из перечисленных методов диагностики выдают изображение в качестве результата. Это рентгенография и компьютерная томография (далее КТ). В следующем разделе будут рассмотрены основные плюсы и минусы данных методов

Рентгенография грудной клетки позволяет получить изображение органов грудной полости посредством облучения рентгеновскими лучами. В зависимости от плотности тканей, через которые проходят рентгеновские лучи, эти ткани по-разному изображаются на снимке. Более плотные ткани выглядят как более светлые участки. Таким образом, альвеолы, содержащие жидкость из-за пневмонии, выглядят как светлые участки, тогда как здоровые участки легких отображаются как затемненные области.

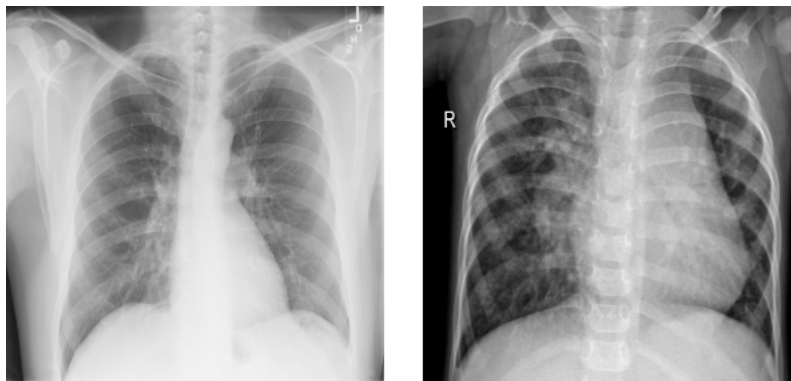


Рис. 1: Рентгенограммы легких, здоровые легкие (слева) и пораженные пневмонией (справа).

плюсы:

- достаточно качественное изображение (хоть и менее информативное, чем у КТ)
- проведение процедуры не требует дополнительной подготовки
- более широкая (по сравнению с аппаратами КТ) распространенность в лечебных учреждениях.

минусы:

- для проведения процедуры требуется, чтобы пациент вдохнул и задержал дыхание. Обычно это не является проблемой, за исключением случаев, когда пациент, например, находится в бессознательном состоянии.
- зашумленность изображения. Особенно явно это проявляется, например, при обследовании людей с большим количеством жировой тка-

ни, это обстоятельство приводит к тому, что весь снимок оказывается значительно более светлым, чем обычно.

- довольно высокие требования к квалификации медицинского персонала из-за зашумленности изображений

Принцип действия КТ основан на получении большого числа снимков легких пациента с помощью датчика с рентгеновским излучателем, который движется по спирали вокруг тела человека. Таким образом, путем съемки с разных ракурсов, получаются снимки послойного разделения легких, число снимков - почти тысяча, каждый из которых содержит изображение легочной ткани толщиной около 1 мм.

Затем снимки передаются на компьютер, на котором из множества снимков формируются изображения в разных плоскостях и трехмерная модель орган, позволяющая детально изучить состояние всех легочных отделов.

плюсы:

- значительно более качественное (по сравнению с рентгенограммой) изображение
- проведение процедуры не требует дополнительной подготовки

минусы:

- значительная доза облучения, что исключает использование процедуры для беременных женщин и детей, за исключением критических случаев.
- дороговизна аппаратов, что приводит к тому, что их нет в большинстве лечебных учреждений.

Пример КТ легких приведен на Рис. 2:

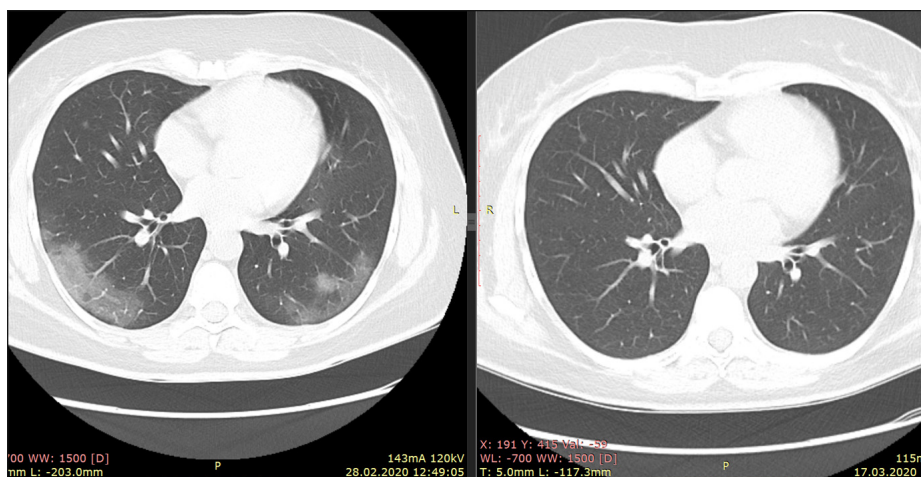


Рис. 2: Пример КТ легких.

Подводя итог вышесказанному, можно сказать, что разработка программного обеспечения для распознавания пневмонии является актуальной задачей. В качестве источника медицинских данных в форме изображений были выбраны рентгенограммы легких из-за распространенности рентгеновских аппаратов.

1.3 Описание задачи классификации.

Задача может быть поставлена следующим образом: пусть имеется некоторое конечное множество объектов (X), каждый из которых относится к одному из нескольких классов. Классовая принадлежность некоторого подмножества множества X известна. Обозначим это подмножество как обучающее множество X_{train} . Требуется создать алгоритм, позволяющий классифицировать, то есть указать номер класса, к которому принадлежит объект, произвольный $x \in X \setminus X_{train}$.

Построенный алгоритм должен подчерпнуть ключевые закономерности, позволяющие отнести объект к одному из классов, из элементов обучающей выборки.

Классификация относится к задаче обучения с учителем: объектам обучающего множества заранее присвоены метки классов. Под учителем следует подразумевать субъекта (человека или, возможно, другой алгоритм), который сопоставил объекты обучающей выборки и классы, к которым они принадлежат.

Отдельно можно рассмотреть обучение без учителя, иными словами, в множестве X нет объектов, которым присвоены метки классов. И объекты могут быть классифицированы только на основании их степени сходства друг с другом. Однако такой тип задач принято называть задачами кластеризации.

Объекты характеризуются набором признаков, которые можно поделить на следующие типы:

- Бинарные признаки: признак может принимать значения из множества $\{0; 1\}$
примерами данного типа признаков могут служить ответы на вопросы: "Доход клиента выше среднего по городу?" или "Цвет этого яблока - зеленый?".
- Вещественные признаки: значение является вещественным числом ($\in R$)
примеры: Возраст, площадь квартиры.
- Категориальные признаки: представляют собой неупорядоченное множество значений, которые могут только проверяться на равенство, но не могут сравниваться в контексте "больше" или "меньше"
примеры: цвет глаз человека, город проживания.
- Порядковые признаки: значения принадлежат к дискретному, упорядоченному множеству.
примеры: тип населенного пункта, уровень образования человека

Рассмотрим способ представления объектов в векторной форме. Если j -ый признак объекта $x \in X$ представлен значением d_j и количество признаков $= n$, то объект x может быть представлен следующим образом:

$$x = (d_1, d_2, \dots, d_n).$$

Допустим, все признаки объекта x являются вещественными, тогда x может быть представлен как точка n -мерного евклидова пространства.

Рассмотрим простейший случай классификации объектов, имеющих два вещественных признака и принадлежащих двум классам.

На Рис 3. изображено двумерное пространство признаков d_1 и d_2 . Объекты двух классов, являющиеся точками этого пространства, разделены нелинейной границей между классами.

Алгоритм-классификатор должен наилучшим образом аппроксимировать данную границу.

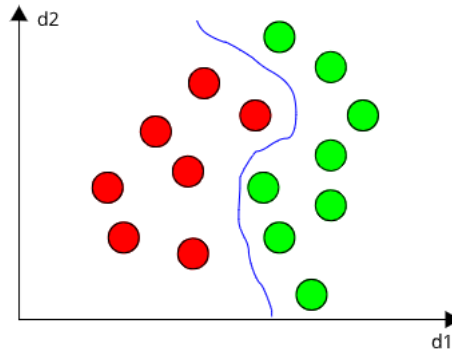


Рис. 3: Пример распределения объектов двух классов.

1.4 Распознавание пневмонии как задача бинарной классификации.

Построение алгоритмов, осуществляющих классификацию рентгенограмм легких, можно отнести к задаче обучения классификатора с учителем.

Так как рентгенограммы являются черно-белыми изображениями, каждый пиксель снимка может быть представлен всего одним целым числом, которое может быть преобразовано в вещественное число путем нормализации.

Таким образом каждое изображение может быть представлено в виде вектора, состоящего из вещественных чисел, размерности $1 \times h * w$ или матрицы размерности $h \times w$, под h понимается высота изображения в пикселях, а под w - его ширина.

Отдельный интерес представляет поиск подходящего наборов данных, содержащего достаточное количество размеченных снимков легких. В качестве примера такого набора можно привести [15]. В этом датасете

содержатся 112120 снимков грудной клетки, принадлежащие 30805 различным людям. Снимки относятся к 14-ти различным болезням (в том числе пневмонии), причем некоторые снимки принадлежат людям, болеющим несколькими болезнями.

альтернативой ему может служить [14] Этот датасет только снимки легких, пораженных пневмонией, и снимки легких условно здоровых (без видимых на снимках патологий) людей.

Хотя первый датасет представляет собой значительно более комплексный и тщательно подготовленный набор снимков, для решения задач, поставленных в данной ВКР, использовался второй вариант, так как в этом наборе снимки поделены на две интересующие нас категории, содержащие достаточное количество примеров.

Разрабатываемые в рамках ВКР программы - классификаторы решают задачу бинарной классификации. Ответом алгоритма является вещественное число $ans \in [0, 1]$, которое можно интерпретировать как вероятность того, что полученное на входе изображение содержит рентгеновский снимок легких человека, больного пневмонией. Соответственно предсказанная вероятность того, что на снимке изображены легкие без признаков пневмонии равна $1 - ans$.

Так как постановка задачи классификации требует того, чтобы на выходе алгоритма произвольному входному изображению присваивается метка класса, мы можем относить к классу "пневмония" те снимки, для которых $ans \geq 0.5$ и к классу "не пневмония" те снимки, для которых $ans < 0.5$.

Глава 2. Описание используемых подходов.

2.1 1. Подход, основанный на использовании сверточных нейронных сетей.

Искусственные нейронные сети являются инструментом, с помощью которого можно попытаться иммитировать процесс обучения и исправления ошибок, протекающий в биологических нервных системах. Таким образом, ИНС представляют собой множество моделей, использующих такие же принципы организации и функционирования, как и те, что имеют место в нервных системах живых организмов.

Искусственный нейрон является составным компонентом ИНС и может быть описан следующим образом: входные сигналы $x_j, j \in [1, \dots, n]$ (могут быть как исходными данными, так и выходными значениями других нейронов сети) подаются через n соединений, каждое из которых ассоциировано с весом $w_j \in R, j \in [1, \dots, n]$. Помимо соединений, передающих входные сигналы, каждый нейрон имеет значение смещения $b \in R$. Внутри нейрона вычисляется взвешенная сумма входных сигналов и смещения $s = (\sum_{j=1}^n w_j * x_j) + b$ далее к этой сумме применяется функция активации $f(s)$, связанная с данным нейроном. Возвращаемое этой функцией значение является выходным сигналом нейрона.

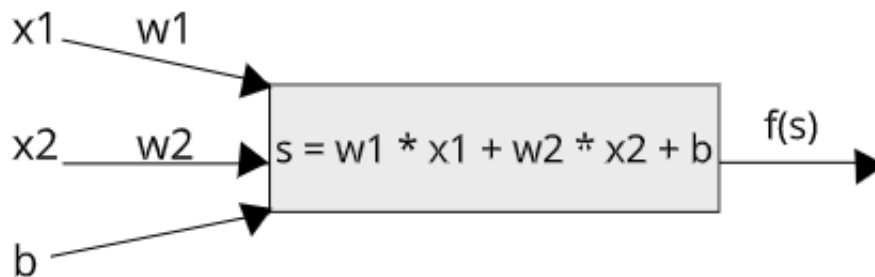


Рис. 4: Искусственный нейрон.

Обычно нейронная сеть состоит из множества нейронов, объединенных в слои таким образом, что выходные значения одного слоя являются входными сигналами для нейронов следующего слоя.

2.1.1 Функции активации.

Функцию активации можно воспринимать как способ нормализации выходного значения нейрона. Таким образом, если вычисленная внутри нейрона сумма s является некоторым большим числом, то $f(s)$ вернет значение, заключенное в нужном диапазоне. Существует много функций активации, поэтому далее будут рассмотрены только основные:

Линейная $f(s) = k * s$, где k - константа. Функции такого типа редко используются, так как при применении линейной функции градиент не зависит от входных данных и является константой, что делает невозможным применение метода обратного распространения ошибки.

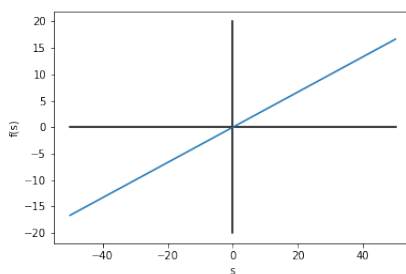


Рис. 5: Линейная функция при $k = 1/3$.

Логистическая сигмоида (sigmoid) $f(s) = \frac{1}{1+exp(-s)}$ является нелинейной S-образной функцией и широко используется в большом количестве сетей. Как можно заметить, при $s = 0$ $f(s)$ не равна нулю, что нежелательно, так как может привести к зигзагообразным скачкам при обновлении весов во время градиентного спуска. Вторым существенным недостатком является затухание градиента, которое выражается в том, что при больших положительных или отрицательных значениях s , график функции становится почти параллельным оси абсцисс, это приводит к тому, что градиент функции в этих точках близок к нулю, таким образом, эффективное применение метода обратного распространения ошибки становится невозможным.

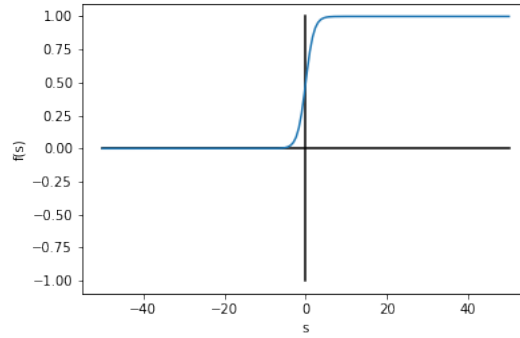


Рис. 6: sigmoid

Гиперболический тангенс (\tanh) $f(s) = \frac{\exp(2s)-1}{\exp(2s)+1}$. Так же как и сигмоида, является S-образной функцией, но в отличие от нее, центрирован в нуле. Проблема затухающего градиента имеет место и для \tanh , хоть и в значительно меньшей степени.

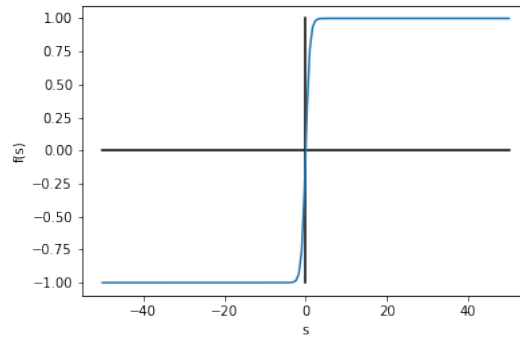


Рис. 7: tanh.

Rectified Linear Unit (ReLU) $f(s) = \max(0, s)$. Эта нелинейная дифференцируемая функция активации является одной из самых широко используемых в глубоких нейронных сетях.

Одним из преимуществ этой функции можно назвать простоту вычисления производной: для отрицательных значений $s - 0$, для положительных $- 1$. Еще одним преимуществом является разреженность активации: при использовании такой функции, как \tanh , активируются почти все нейроны, это может негативно отразиться на качестве обучения сети. При использовании ReLU число активируемых нейронов будет существенно меньше из-за характеристик функции.

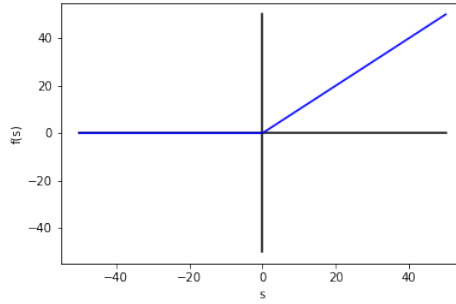


Рис. 8: ReLU.

Softmax представляет собой обобщение логистической функции для многоклассовой классификации. Функция преобразует вектор in размерности M в вектор out такой же размерности, где каждая компонента выходного вектора представлена вещественным числом в интервале $[0, 1]$ и сумма координат равна 1. j -ая компонента вектора out вычисляется следующим образом: $out_j = \frac{\exp(in_j)}{\sum_{k=1}^M \exp(in_k)}$

2.1.2 Функция потерь.

Пусть X - множество описаний объектов, а Y - множество возможных ответов. Предположим, что существует неизвестное нам отображение $X \rightarrow Y$, причем нам известны правильные ответы $y \in Y$ для некоторой конечной подвыборки из X . Обозначим эту подвыборку как обучающее множество $T = \{(x_1, y_1), \dots, (x_n, y_n)\}$, состоящее из пар (x_j, y_j) , где x_j - описание некоего объекта, а y_j - известный ответ для этого объекта.

Допустим, для объекта x_j был дан ответ y_j^* . Функция, характеризующая отклонение ответа нейронной сети от известного правильного ответа y_j называется функцией потерь $Loss(y_j, y_j^*)$. Задача обучения нейронной сети сводится к задаче минимизации функционала качества $Q = \frac{1}{n} \sum_{j=1}^n Loss(y_j, y_j^*)$ путем изменения весов сети. Далее будут рассмотрены наиболее популярные функции потерь.

Среднеквадратичная ошибка (MSE): $Loss(y_j, y_j^*) = ||y_j - y_j^*||^2$ является наиболее распространенной функцией потерь. Данная функция широко используется в линейной регрессии для расчета эффективности.

Категориальная кросс-энтропия: Пусть решается задача многоклассовой классификации и количество классов равно K . Ответ сети y_j^* в данном случае представляет собой вектор размерности K , каждая q -ая компонента вектора определяет вероятность того, что объект x_j принадлежит классу q , $\sum_{q=1}^K y_q^* = 1$. Верный ответ y тоже является вектором размерности K , причем только одна компонента этого вектора равна 1, все остальные равны 0. Категориальная кросс-энтропия измеряет расхождение вероятностных распределений y_j^* и y_j : $Loss(y_j, y_j^*) = -\sum_{q=1}^K y_{jq} * \log(y_{jq}^*)$

Бинарная кросс-энтропия: $Loss(y_j, y_j^*) = -y_j * \log(y_j^*) - (1 - y_j) * \log(1 - y_j^*)$. Является частным случаем категориальной кросс-энтропии при $K = 2$. Ответ нейронной сети для объекта x_j представляет собой единственное вещественное число y_j^* , равное вероятности принадлежности объекта x_j классу 1, вероятность принадлежности к классу 2 равна $1 - y_j^*$.

Категориальная кросс-энтропия обычно используется когда в качестве функции активации последнего слоя сети применяется softmax. Бинарная кросс-энтропия как правило используется совместно с функцией активации sigmoid. Стоит отметить, что для использования кросс-энтропии выходное значение (или каждая из компонент выходного вектора) должно принадлежать отрезку $[0, 1]$. Softmax и sigmoid подходят под это требование.

2.1.3 Переобучение.

Переобучение - это явление, в результате которого нейронная сеть избыточно настраивается под обучающее множество, теряя способность к обобщению признаков объектов. Это приводит к тому, что сеть показывает хорошие результаты на обучающем множестве, но на тестовых объектах, которые не входят в обучающее множество, результаты работы сети значительно хуже. Далее будут рассмотрены несколько способов борьбы с переобучением.

Регуляризация. Идея заключается в том, чтобы уменьшить значения весов, что позволяет упростить модель нейронной сети, так как слишком сложная и более гибкая сеть скорее всего будет максимально настраиваться под обучающее множество. Отдельно стоит рассмотреть L1 и L2 регуляризацию. В обоих случаях к функционалу качества $Q = \frac{1}{n} \sum_{k=1}^n Loss(y_j, y_j^*)$ добавляется дополнительное слагаемое. В случае L1 регуляризации это слагаемое имеет вид $\lambda_1 \sum_w |w|$ и равно сумме абсолютных значений всех весов в нейронной сети. Для L2 регуляризации абсолютные значения заменяются квадратами весов: $\lambda_2 \sum_w w^2$ константы λ_1 и λ_2 называются параметрами регуляризации.

Dropout. Данный метод регуляризации заключается в том, что в одном или нескольких слоях нейронной сети каждый нейрон отключается с заданной вероятностью p . Это означает, что сигналы полученные на вход и вычисленный на их основе выходной сигнал нейрона игнорируется на текущем этапе обучения. Такой прием уменьшает степень взаимoadaptации нейронов, что препятствует переобучению. Во время запуска нейронной сети на тестовом множестве, как и при запуске на реальных данных, никакие нейроны не отключаются, но выходной сигнал каждого нейрона умножается на $q = 1 - p$ - вероятность, что нейрон не был отключен во время обучения.

Аугментация данных - техника, позволяющая искусственно создать

дополнительные объекты обучающего множества из уже имеющихся. При работе с изображениями новые тренировочные объекты могут быть получены путем совершения ряда случайных трансформаций над объектами исходного множества таких как поворот на некоторый угол, смещение, увеличение / уменьшение масштаба изображения, отражение по вертикали / горизонтали и так далее. Можно выделить два вида аугментации данных.

Оффлайн аугментация применяется на стадии препроцессинга. Путем описанных выше трансформаций получают дополнительные изображения и добавляются к объектам исходного множества. После этого начинается процесс обучения, проходящий в течение некоторого количества эпох обучения (обозначим это число как $epochs$). При применении оффлайн аугментации каждый снимок обучающего множества, как принадлежащий исходному множеству, так и полученный в результате трансформаций, будет подан на вход сети $epochs$ раз.

Под эпохой обучения понимается совокупность операций, в результате которой нейронной сети подаются все объекты из обучающего множества. Каждая эпоха состоит из нескольких итераций. На каждой итерации нейронной сети подается на вход некоторое число примеров из обучающего множества, обозначим это число как $batch\ size$, после запуска на данном числе примеров методом обратного распространения ошибки вычисляется градиент и обновляются веса нейронной сети. Таким образом число итераций в одной эпохе равно $\left\lceil \frac{|T|}{batchsize} \right\rceil$ где $|T|$ - количество элементов в обучающем множестве.

Альтернативной техникой является так называемая онлайн или real-time аугментация данных. Отличие заключается в том, что количество объектов в обучающем множестве не изменяется. Вместо этого каждый элемент обучающего множества трансформируется перед каждой эпохой обучения. Таким образом, сеть не "видит" каждый из объектов более одного раза. Этот факт значительно улучшает обобщающую способность сети[16].

2.1.4 Сверточные нейронные сети.

Сверточная нейронная сеть (СНС) на сегодняшний день являются одной из самых популярных и эффективных для решения задач распознавания изображений.

Работа СНС представляет собой поэтапный переход от конкретных особенностей изображения к более абстрактным деталям, затем к ещё более абстрактным деталям и так далее вплоть до выделения понятий высокого уровня. Эти абстрактные признаки формируют так называемые карты признаков (фактически, матрицы). При этом сеть самонастраивается и вырабатывает необходимую иерархию абстрактных признаков (последовательности карт признаков), отбрасывая маловажные детали и выделяя существенные.

Идея сверточных сетей заключается в чередовании слоев свертки и субдискретизирующих слоев (слоев подвыборки). Сеть имеет многослойную структуру, обучение производится стандартными методами, обычно методом обратного распространения ошибки.

Сверточные слои. Формально свёртка представляет собой специфическую операцию умножения исходной матрицы на матрицу меньшего размера, называемую ядром свертки или фильтром. Ядро как бы «двигается» по исходной матрице слева направо и сверху вниз и в каждом положении считается скалярное произведение фильтра и той части исходной матрицы, на которую он наложен. Получившееся число запоминается как соответствующий элемент результата. Сверточный слой представляет собой применение операции свертки к выходам с предыдущего слоя, а веса ядра свертки являются обучаемыми параметрами.

На Рис. 9 изображено вычисление результата свертки(справа) для элемента матрицы, расположенного на пересечении 2-го столбца и 2-ой строки (слева) ядром свертки размерности 3×3 (посередине). $1 * 2 + 0 * 0 + 2 * 1 + 2 * 0 + 2 * 3 + 4 * 1 + 0 * 1 + 2 * 2 + 1 * 1 = 2 + 0 + 2 + 0 + 6 + 4 + 0 + 4 + 1 = 19$

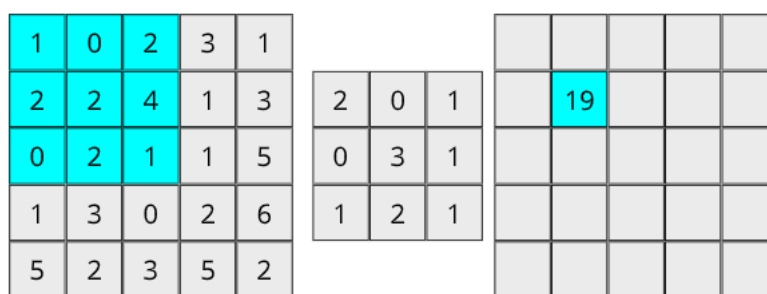


Рис. 9: Операция свертки.

Отдельно стоит рассмотреть варианты обработки элементов, расположенных на краях матрицы, так как в этих случаях часть ядра свертки накладывается на несуществующие элементы. Наиболее популярным подходом является дополнение исходной матрицы нулевыми элементами вдоль границ матрицы.

На Рис. 10 изображен процесс вычисления результата свертки для элемента на пересечении первой строки и первого столбца. Дополнительные нулевые элементы и накладываемые на них элементы ядра свертки закрашены зеленым.

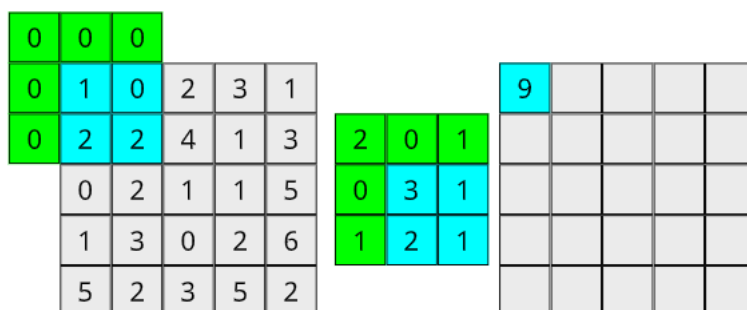


Рис. 10: Операция свертки для элемента на границе матрицы.

После вычисления всех элементов результирующей матрицы к каждому из этих элементов применяется функция активации, ассоциированная с данным сверточным слоем (например ReLU). Сверточный слой как правило содержит несколько ядер свертки. Таким образом, после прохождения сверточного слоя из исходной (исходных) матриц может получиться большее количество матриц, порожденных из исходной различными ядрами свертки.

Слой субдискретизации. Слои этого типа служат для уменьшения размерности (как правило в несколько раз) исходной матрицы (матриц). Это возможно осуществить различными методами, но довольно часто используется метод выбора максимального элемента из блоков, на которые разделена исходная матрица.

На Рис. 11 изображена операция подвыборки, применяемая к исходной матрице размерности 4×4 , размерность блоков, на которые разделена матрица равна 2×2 . Блоки на рисунке отмечены различными цветами.

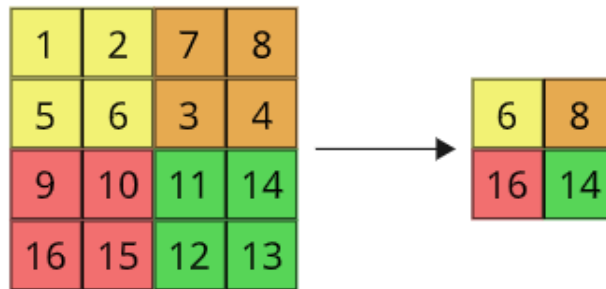


Рис. 11: Операция подвыборки.

Как и в случае со сверточными слоями, присутствует проблема обработки элементов на границе матрицы. Например, если число столбцов и/или строк матрицы нечетное, а блоки подвыборки имеют размерность 2×2 , останутся элементы, которые придется обрабатывать отдельно. Наиболее популярным способом является дополнение матрицы справа и снизу элементами, содержащими такое значение, что эти элементы никогда не будут выбраны функцией подвыборки, для определенности обозначим это значение как *inf*. Если функция подвыборки выбирает максимальный элемент в блоке в качестве *inf* можно взять некоторое отрицательное число.

На Рис. 12 изображен описываемый случай.

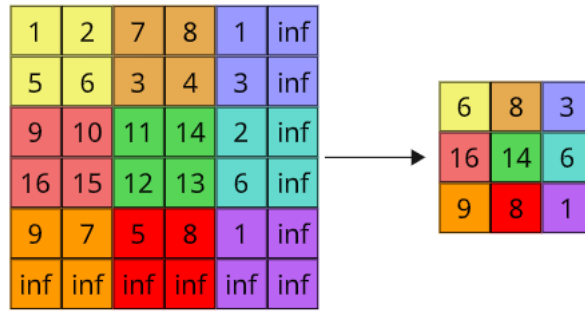


Рис. 12: Операция подвыборки для случая, когда размерность матрицы не кратна размерности блока подвыборки.

Слой подвыборки выдает столько же матриц, сколько принял, изменяется только размерность.

Полносвязные слои. На выходе последовательности свёрточных слоёв (свёртки и субдискретизации) получается набор конечных карт признаков. Он, в свою очередь, подается на вход полносвязному участку нейронной сети, который тоже может состоять из нескольких слоёв. После чего сигнал подается на выходной слой, на выходах которого формируется результат (отклик сети на входное воздействие). Полносвязный слой характеризуется тем, что каждый нейрон этого слоя соединен со всеми нейронами предыдущего слоя. Каждая связь, соединяющая нейроны, имеет свой вес.

Пример полносвязного слоя изображен на Рис. 13. Выходные сигналы предыдущего слоя обозначены как x_j , веса связей между j - ым сигналом предыдущего слоя и k - ым нейроном полносвязного слоя обозначается как w_{jk} . Функция активации, формирующая выходной сигнал слоя обозначена как $f(s)$.

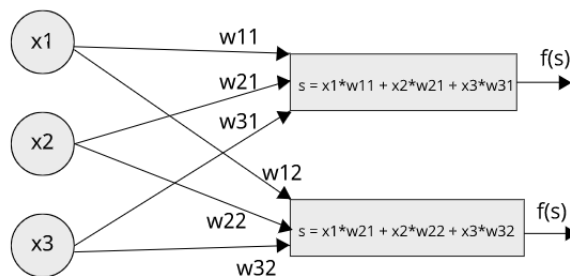


Рис. 13: Пример полносвязного слоя.

При решении задач классификации участок сети, содержащий сверточные слои и слои подвыборки отвечает за выделение более важных признаков и отбрасывание менее важных. Непосредственно классификацию производит полносвязный участок (а точнее его последний слой), на вход которому подается вектор значительно меньшей размерности чем тот, что подавался на вход сверточному участку.

2.1.5 Структура нейронной сети, решающей задачу распознавания пневмонии.

Идея использовать сверточную нейронную сеть для распознавания пневмонии не нова, данная задача решается например в [4-7]. В рамках ВКР, во многом опираясь на вышеуказанные источники, автор предлагает для решения задачи распознавания пневмонии сверточную нейронную сеть, имеющую следующую структуру:

1. входной сверточный слой, количество фильтров = 32, ядро свертки размерности 3×3 ,
шаг ядра свертки = 1, функция активации - ReLU
2. субдискретизирующий слой, размер блока подвыборки 2×2 ,
шаг блока подвыборки = 2
3. сверточный слой, количество фильтров = 64, ядро свертки размерности 3×3 ,
шаг ядра свертки = 1, функция активации - ReLU
4. слой Dropout, вероятность отключения нейронов предыдущего слоя = 0.1
5. субдискретизирующий слой, размер блока подвыборки 2×2 ,
шаг блока подвыборки = 2
6. сверточный слой, количество фильтров = 64, ядро свертки размерности 3×3 ,
шаг ядра свертки = 1, функция активации - ReLU

7. слой Dropout, вероятность отключения нейронов предыдущего слоя = 0.1
8. субдискретизирующий слой, размер блока подвыборки 2×2 , шаг блока подвыборки = 2
9. сверточный слой, количество фильтров = 128, ядро свертки размерности 3×3 , шаг ядра свертки = 1, функция активации - ReLU
10. слой Dropout, вероятность отключения нейронов предыдущего слоя = 0.2
11. субдискретизирующий слой, размер блока подвыборки 2×2 , шаг блока подвыборки = 2
12. сверточный слой, количество фильтров = 256, ядро свертки размерности 3×3 , шаг ядра свертки = 1, функция активации - ReLU
13. слой Dropout, вероятность отключения нейронов предыдущего слоя = 0.2
14. субдискретизирующий слой, размер блока подвыборки 2×2 , шаг блока подвыборки = 2
15. слой Flatten, преобразующий результирующие матрицы предыдущего слоя в одномерный вектор.
16. полносвязный слой, количество нейронов = 128, функция активации - ReLU
17. слой Dropout, вероятность отключения нейронов предыдущего слоя = 0.2
18. полносвязный слой, количество нейронов = 1, функция активации - sigmoid

Задача распознавания пневмонии по рентгеновским снимкам является задачей бинарной классификации. Поэтому последний полносвязный слой нейронной сети, решающей эту задачу, состоит из всего одного нейрона с функцией активации `sigmoid`. В качестве функции потерь была выбрана бинарная кросс-энтропия.

Код программы написан на языке программирования Python с использованием открытой нейросетевой библиотекой Keras. Модель нейронной основывается на классе `keras.Sequential`. Далее будут перечислены классы, инкапсулирующие поведение различных типов слоев нейронной сети:

1. Сверточные слои: `keras.layers.Conv2D`
2. Субдискретизирующие слои: `keras.layers.MaxPool2D`
3. Слой Dropout: `keras.layers.Dropout`
4. Слой Flatten: `keras.layers.Flatten`
5. Полносвязный слой: `keras.layers.Dense`

Конструктор первого(входного) сверточного слоя принимает дополнительный аргумент - кортеж с именем `input_shape`, первые две компоненты отвечают за количество пикселей вдоль осей y и x входного изображения (в рамках программы размерность задается глобальной переменной `img_size`), третья компонента определяет число цветовых каналов, так как изображения черно-белые, эта компонента = 1

Перед процессом обучения исходные изображения $x \in X$ и сопоставленные этим изображениям экспертные решения $y \in Y$ делятся на три непересекающихся множества: тренировочное (`x_train`, `y_train`), валидационное (`x_val`, `y_val`) и тестовое (`x_test`, `y_test`). Все изображения, возможно имеющие разный размер, приводятся к размеру `img_size` \times `img_size`. После чего числовые представления изображений нормируются. Так как рентгенограммы легких - черно-белые изображения, каждый пиксель представляется целым числом из отрезка $[0, 255]$, где значение 0 обозначает черный цвет, а 255 - белый. Нормирование происходит путем деления числового представления каждого пикселя на максимально возможное значение

(255), после чего каждый пиксель представляется рациональным числом из отрезка $[0, 1]$

Обучение нейронной сети производилось с использованием real-time аугментации данных, для этих целей использовался класс `keras.preprocessing.image` трансформации над обучающим множеством были задано как:

- Поворот на случайный угол из отрезка $[-20; 20]$ градусов
- Смещение по вертикали и или горизонтали на 10% от высоты/ширины изображения.
- Изменение масштаба изображения. Множитель масштаба $\in [0.9; 1.1]$

В качестве оптимизатора обучения был выбран RMSProp, размер батча равен 32.

2.2 Подход, основанный на использовании колмогоровской сложности.

2.2.1 Основные теоретические аспекты.

Дадим определение Колмогоровской сложности в соответствии с [11].

Способом описания, или декомпрессором, называется произвольное вычислимое частичное отображение D из множества двоичных слов Ξ в себя. (Вычислимость отображения D означает, что есть алгоритм, который применим к словам из области определения отображения D и только к ним; результат применения алгоритма к слову x есть $D(x)$.) Если $D(y) = x$, говорят, что y является описанием x при способе описания D . Для каждого способа описания D мы определяем сложность относительно этого способа описания, полагая ее равной длине кратчайшего описания.

$$KS_D(x) = \min\{l(y) | D(y) = x\}.$$

При этом минимум пустого множества считается равным ∞ . Говорят, что способ описания D_1 не хуже способа описания D_2 , если найдется такая константа c , что

$$KS_{D1}(x) \leq KS_{D2}(x) + c.$$

для всех слов x . Способ описания называют оптимальным, если он не хуже любого другого способа описания.

Теперь мы можем зафиксировать некоторый (не обязательно оптимальный) способ описания и будем обозначать сложность слова x относительно данного способа описания как $K(x)$. В рамках данной статьи в качестве отображения D мы будем использовать стандартные программы-архиваторы, а сложностью двоичного слова x будем считать количество байтов в его сжатой версии.

Пусть y - еще одно двоичное слово. Обозначим минимальное количество битов, необходимых для восстановления x из y как $K(x|y)$. Для любой пары слов x и y можно определить нормализованное расстояние сжатия как

$$NCD(x, y) = \frac{\max\{K(x|y), K(y|x)\}}{\max\{K(x), K(y)\}}.$$

Таким образом, мы считаем два объекта близкими, если мы можем значительно сжать один из них, учитывая информацию, содержащуюся в другом.

В [12] доказывается теорема о симметрии алгоритмической информации, из которой следует приближительное равенство

$$K(x|y) \approx K(yx) - K(y).$$

Под yx мы понимаем конкатенацию двоичных строк y и x

Так как на практике $K(xy) \approx K(yx)$, мы можем вычислять NCD как

$$NCD(x, y) = \frac{K(xy) - \min\{K(x), K(y)\}}{\max\{K(x), K(y)\}}. \quad (1)$$

2.2.2 Алгоритм классификации.

Пусть исходное множество поделено на два непересекающихся множества:

первое - "обучающее" множество снимков $Y = Y_P \cup Y_N$, $|Y_P| = |Y_N| = M$, где Y_P - подмножество снимков с признаками пневмонии, а Y_N - содержит снимки легких без признаков пневмонии. Название данного множества дано в кавычках, так как никакого обучения в смысле обучения нейронных сетей не производится. В то же время это название должно подчеркнуть схожесть выполняемых данным множеством функций. И нейронные сети, и программы, использующие колмогоровскую сложность (далее ПКС) извлекают некоторые закономерности из обучающего множества и использовать эти закономерности для классификации ранее не встречавшихся изображений.

второе множество - тестовое, обозначим его как X , объектам данного множества, так же как и объектам множества Y , сопоставлены метки классов P или N. Тестовое множество служит для измерения качества классификации.

Подготовительный этап:

1. Преобразовать каждый из снимков множества Y в текстовый файл, содержащий двоичную строку. Таким образом мы получаем

$$y_{P_j}, j \in \{1, \dots, M\}$$

и

$$y_{N_k}, k \in \{1, \dots, M\}$$

2. заархивировать каждый из текстовых файлов, полученных в шаге 1, выбранной программой-архиватором; размер заархивированной версии соответствующего текстового файла равен Колмогоровской сложности двоичной строки, содержащейся в файле. Таким образом мы получаем.

$$K(y_{P_j}), j \in \{1, \dots, M\}$$

и

$$K(y_{N_k}), k \in \{1, \dots, M\}$$

Основной этап: Пусть после подготовительного этапа на вход программе подается объект из тестового множества X .

4. Повторить для этого изображения действия, описанные в шагах 1 и 2, получив x и $K(x)$ соответственно.
5. Для каждого $y = y_{P_j}(y_{N_j}), j \in \{1, \dots, M\}$:
 - (а) вычислить $K(xy)$ путем создания текстового файла, образованного конкатенацией x и y , и последующей архивации этого файла.
 - (б) вычислить $NCD(x, y)$ по формуле (1).

Совершив вышеописанные действия для каждого из классов, мы получаем два вектора $1 \times M$, элементы которых содержат расстояния от x до соответствующих элементов Y_P и Y_N

6. сравнить арифметические средние от значений векторов, отнести x к классу с наименьшим результатом

2.2.3 Детали реализации алгоритма классификации.

При реализации алгоритма, использующего колмогоровскую сложность (далее - ПКС), необходимо обратить внимание на следующие детали:

Размер снимков: Очевидно, что для корректной работы алгоритма снимки из множества Y и классифицируемый снимок должны быть одного размера. Кроме того, мы можем рассматривать размер снимка в качестве гиперпараметра, в некоторой степени влияющего на качество классификации. Различные результаты работы программы КС на снимках разного размера будут приведены в третьей главе

Способ представления снимка в виде двоичной строки: Так как рентгенограммы представляют собой черно-белые изображения, каждый пиксель может быть представлен двоичной строкой из 8 символов. Следова-

тельно, двоичное представление можно получить, заменяя числовое значение каждого пикселя его двоичной записью и конкатенируя эти записи, двигаясь по снимку слева-направо сверху-вниз. При классификации цветных изображений каждый пиксель кодируется тремя байтами (rgb) или четырьмя (rgba). В данном случае должны быть рассмотрены альтернативные способы представления изображения в виде двоичной строки.

Используемая программа архиватор. В статье [17] рассматриваются такие программы-архиваторы как:

- bzip2
- gzip
- XZ
- 7z
- ACE
- ARC
- PAQ
- RAR
- ZIP

для реализации метода, используемого в данной ВКР, был выбран архиватор 7z.

Количество элементов в множестве Y . Можно предположить, что чем больше примеров содержит "обучающее" множество, тем более качественной будет классификация.

В данной работе это предположение не проверялось из-за связанных с увеличением мощности множества Y вычислительных затрат. Кроме того, возможность довольно высокого качества классификации при неболь-

шом обучающем множестве является отличительным свойством и одним из главных преимуществ ПКС.

Эта особенность будет подробно описана в следующей главе, посвященной сравнению результатов.

Глава 3. Сравнение результатов.

3.1 Описание данных.

Для практической реализации используемых алгоритмов и проведения экспериментов использовался открытый набор данных [15].

Набор содержит 5851 размеченных рентгеновских снимков различного размера. Среди них 1583 снимка, содержащих снимки легких без признаков пневмонии, и 4268 изображения с диагнозом пневмония.

Снимки были распределены на следующие непересекающиеся множества: количество снимков в множествах будет обозначаться как (P: количество снимков с пневмонией ; N: количество снимков без пневмонии). В дальнейшем мы будем обозначать класс "пневмония" как P и класс "не пневмония" как N.

- Обучающее множество Train (P:3718; N:1033)
- Валидационное множество Val (P:200; N:200)
- Тестовое множество Test (P:350; N:350)

Сверточная нейронная сеть (Далее - СНС) использует эти множества в соответствии с их названиями.

Программа, использующая колмогоровскую сложность (далее - ПКС), использует в качестве обучающего множества (PКС_train) подмножество множества Train, состоящее из 100 объектов каждого класса.

Таким образом $PКС_train \subset Train$, множество PКС_train (P:100; N:100) эквивалентно множеству Y из 2.2.2.

ПКС никак не использует множество Val и использует множество Test таким же образом, как его использует СНС - для итогового измерения качества классификации.

3.2 Метрики.

Для оценки качества алгоритмов бинарной классификации используются различные метрики [18], наиболее часто встречающиеся на практике будут описаны ниже.

Перед рассмотрением особенностей метрик, имеет смысл определить матрицу ошибок для бинарной классификации. Для определенности будем считать, что по строкам этой матрицы расположены ответы классификатора $a(x)$, а по столбцам - экспертные (истинные) решения y .

	$y = P$	$y = N$
$a(x) = P$	True Positive (TP)	False Positive (FP)
$a(x) = N$	False Negative (FN)	True Negative (TN)

К основным метрикам можно отнести следующие:

accuracy является отношением верно распознанных объектов к их общему количеству:

$$accuracy = \frac{TP+TN}{TP+FP+FN+TN}$$

recall (полнота) или sensitivity (чувствительность) доля верно распознанных позитивных объектов

$$recall = \frac{TP}{TP+FN}$$

specificity (специфичность) доля верно распознанных негативных объектов

$$specificity = \frac{TN}{TN+FP}$$

precision (точность), является отношением верно распознанных позитивных объектов к общему количеству позитивных вердиктов классификатора:

$$precision = \frac{TP}{TP+FP}$$

В зависимости от решаемой задачи, прирост по одной метрике ценится значительно выше, чем прирост по другой.

Так, в задачах медицинской диагностики наибольшее значение имеет sensitivity, так как человек, не получивший подходящего лечения, вернется позже, скорее всего с запущенной болезнью. Когда как здоровый человек, направленный на дополнительные обследования, потратит некоторое количество ресурсов лечебного учреждения впустую. Этот тип ошибок несет значительно меньший вред.

Иными словами, гипердиагностика предпочтительнее гиподиагностики.

3.3 Выбор гиперпараметров и сравнение методов.

Одним из гиперпараметров СНС, решающей задачу распознавания пневмонии, является размерность входного изображения (`input_shape`). Экспериментальным образом было замерено качество классификации СНС в зависимости от $\text{input_shape} \in \{150 \times 150, 256 \times 256, 350 \times 350\}$.

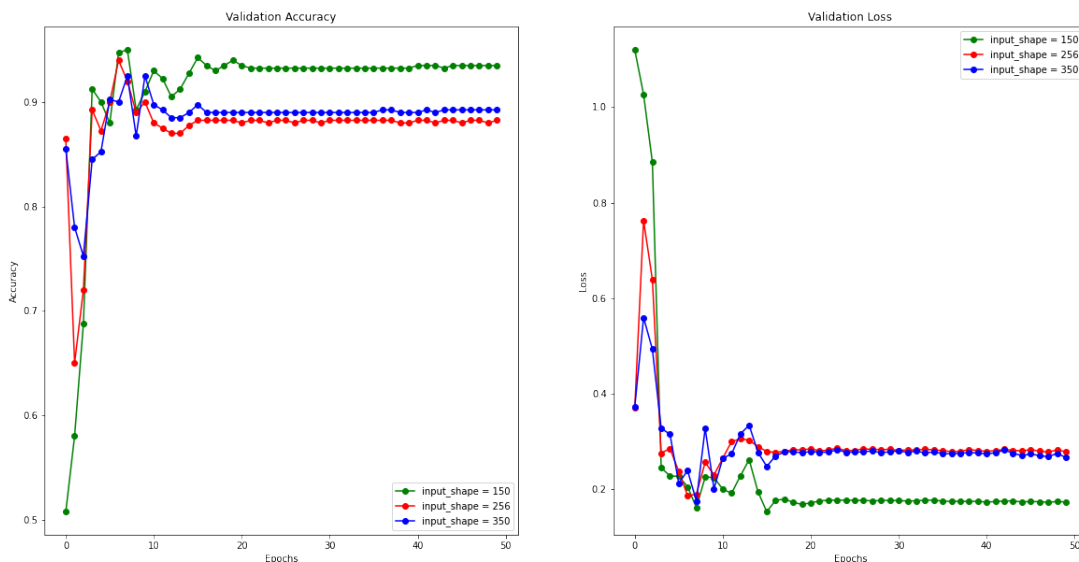


Рис. 14: Результаты на валидационном множестве в зависимости от размера входного изображения.

В результате экспериментов было установлено, что наилучшее качество классификации достигается при `input_shape = 150 × 150`.

После задания оптимальной размерности изображения, Было произведено обучение нейронной сети. Обучение проводилось в течение 70 эпох, после каждой эпохи замерялась доля правильно классифицируемых объектов тренировочного и валидационного множества (training accuracy, validation accuracy) и значения функции потерь (training loss, validation loss). Помимо этих классических метрик, после каждой эпохи обучения для объектов валидационного множества замерялось значение чувствительности (sensitivity) и специфичности (specificity).

На Рис.14 видно, что наилучшие результаты классификации сеть показывала после 16-ой эпохи обучения (выделена на всех трех графиках вертикальной черной линией).

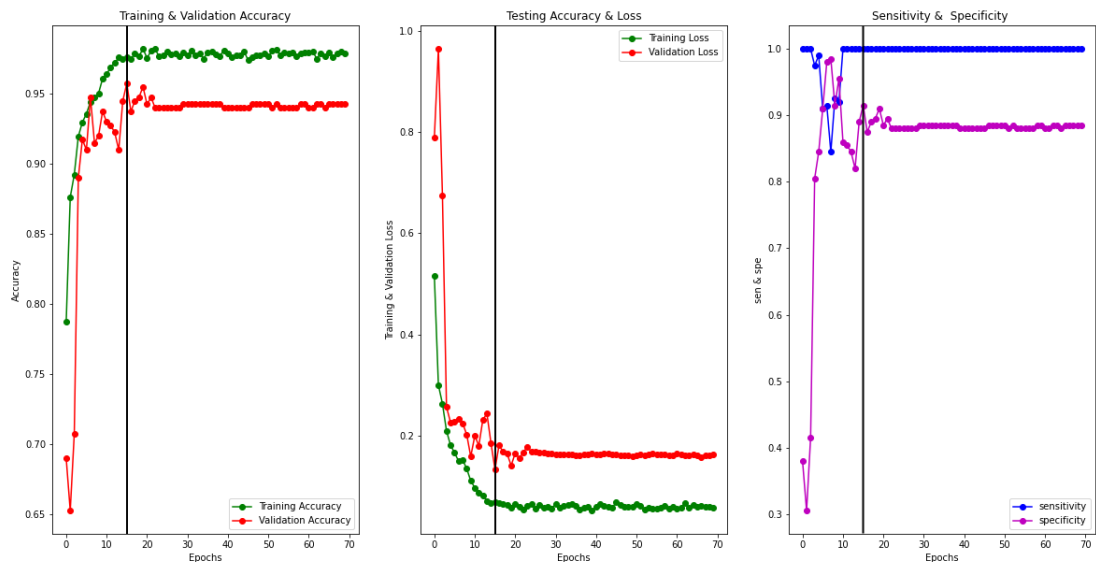


Рис. 15: Процесс обучения нейронной сети.

После 70-ти эпох обучения были получены следующие результаты на тестовом множестве:

epoch = 70	y = P	y = N
a(x) = P	348	94
a(x) = N	2	256

$$sensitivity = 0.99, specificity = 0.73$$

Из-за свойства воспроизводимости результатов при задании так называемого "зерна"(seed) генератора псевдослучайных чисел был произведен запуск обучения при параметрах, полностью повторяющих предыдущий опыт. В этот раз обучение длилось 16 эпох. Результаты получившейся СНС представлены в таблице ниже:

epoch = 16	y = P	y = N
a(x) = P	347	77
a(x) = N	3	273

$$sensitivity = 0.99, specificity = 0.78$$

Для ПКС размерность входного изображения так же является параметром, влияющим на итоговое качество классификации. Далее будет рассмотрено несколько примеров, позволяющих определить оптимальную размерность для ПКС.

input_shape = (128×128)	y = P	y = N
a(x) = P	350	342
a(x) = N	0	8

$$sensitivity = 1.0, specificity = 0.02$$

input_shape = (256×256)	y = P	y = N
a(x) = P	251	280
a(x) = N	99	70

$$sensitivity = 0.71, specificity = 0.2$$

input_shape = (512×512)	y = P	y = N
a(x) = P	323	115
a(x) = N	27	235

$$sensitivity = 0.92, specificity = 0.67$$

input_shape = (650×650)	y = P	y = N
a(x) = P	274	55
a(x) = N	76	295

$$sensitivity = 0.78, specificity = 0.84$$

Таким образом, результаты СНС($sensitivity = 0.99, specificity = 0.78$) лучше результатов ПКС($sensitivity = 0.92, specificity = 0.67$) по каждому из параметров. В связи с этим возникает вопрос: в каких случаях ПКС может оказаться предпочтительнее СНС?

Ответ на этот вопрос кроется в том факте, что для относительно удовлетворительного качества классификации ПКС требует всего (P:100; N:100) размеченных снимков. Когда как для обучения СНС, предположительно, требуется значительно большее количество снимков. Следующий раздел посвящен проверке данного предположения и определении нижней границы мощности обучающего множества, по достижении которой СНС осуществляет более качественную классификацию, чем ПКС.

Приведенные ниже таблицы содержат результаты работы СНС на тестовом множестве при искусственном ограничении обучающего множества Train(P: M; N: M). Число M - количество представителей каждого из классов. Количество эпох во всех тестах = 50, после чего эксперимент повторяется и обучение останавливается на той эпохе, на которой был показан наилучший результат на валидационном множестве.

M = 100	y = P	y = N
a(x) = P	350	350
a(x) = N	0	0

$$sensitivity = 1.0, specificity = 0.0$$

M = 200	y = P	y = N
a(x) = P	350	338
a(x) = N	0	12

$$sensitivity = 1.0, specificity = 0.03$$

M = 250	y = P	y = N
a(x) = P	348	249
a(x) = N	2	101

$$sensitivity = 0.99, specificity = 0.28$$

M = 300	y = P	y = N
a(x) = P	324	57
a(x) = N	26	293

$$sensitivity = 0.92, specificity = 0.83$$

Следует отметить, что наилучший результат, показанный ПКС (sensitivity = 0.92, specificity = 0.67), заключен между результатами СНС при M = 250 (sensitivity = 0.99, specificity = 0.28) и M = 300 (sensitivity = 0.92, specificity = 0.83).

Выводы

На основании результатов, изложенных в главе 3, можно определить основные преимущества и недостатки каждого из подходов. Сверточная нейронная сеть.

преимущества:

- Один из лучших инструментов для классификации изображений
- Наличие открытых библиотек, облегчающих разработку
- Устойчивость к перемещению содержимого входных изображений

недостатки:

- Недостаточно качественная классификация при малом количестве обучающих изображений.

Программа, использующая колмогоровскую сложность. преимущества:

- отсутствие весов, которые необходимо подбирать
- достаточно качественная классификация в случаях, когда мощность обучающего множества мала

недостатки:

- Сравнительно низкое быстродействие из-за многократно вызываемого процесса архивации
- Значительно уступает сверточной нейронной сети, обученной на достаточном количестве примеров.

Заключение

В рамках ВКР все поставленные задачи были выполнены. Была описана задача распознавания пневмонии и методы диагностики этой болезни.

Во второй главе содержатся описания двух подходов к распознаванию пневмонии и предлагаются программные решения, реализующие данные алгоритмы.

Код программ доступен для ознакомления по ссылкам:

<https://github.com/bogdanov978/Pneumonia-classification-CNN>

<https://github.com/bogdanov978/Pneumonia-classification-Kolmogorov-complexity>

В третьей главе было произведено сравнение качества работы программ, классифицирующих примеры изображений из заданного набора данных.

Итогом настоящей работы являются программы: Сверточная нейронная сеть: (sensitivity = 0.99, specificity = 0.78)

Программа, использующая колмогоровскую сложность (sensitivity = 0.92, specificity = 0.67)

Разработанные программы могут быть использованы в качестве программ-консультантов для специалистов-рентгенологов.

Список литературы

- [1] Всемирная организация здравоохранения. Пневмония [Электронный ресурс]. – URL: <https://www.who.int/ru/news-room/fact-sheets/detail/pneumonia> (дата обращения 14.04.2021).
- [2] Струтынский, А.В. Опыт применения компьютерной томографии высокого разрешения легких в диагностике и оценке результатов лечения внебольничной пневмонии / А.В. Струтынский, А.Л. Юдин, Д.В. Семенов, Н.И. Афанасьева, В.А. Семенов, // Пульмонология. – 2014. – №. 4. – С. 68-70.
- [3] Иванов, В.В. и др. Дифференциальная диагностика внебольничной пневмонии и диссеминированного заболевания / В.В. Иванов, М.А. Харитонов, В.В. Данцев, Б.А. Чумак, и др. // Вестник Российской военно-медицинской академии. – 2015. – №. 2. – С. 113-117.
- [4] Ефремцев, В.Г. Классификация рентгеновских изображений грудной клетки больных вирусной пневмонией и COVID-19 с помощью нейронных сетей / В.Г. Ефремцев, Н.Г. Ефремцев, Е.П. Тетерин, П.Е. Тетерин, Е.С. Базавлук // Компьютерная оптика. – 2021. – Т. 45, № 1. – С. 149-153.
- [5] Imran, A. Training a CNN to detect Pneumonia. / A. Imran [Электронный ресурс]. – URL: <https://medium.com/datadriveninvestor/training-a-cnn-to-detect-pneumonia-c42a44101deb> (дата обращения 14.05.2021).
- [6] Chhikara, P. Deep convolutional neural network with transfer learning for detecting pneumonia on chest X-rays / P. Chhikara, P. Singh, P. Gupta, T. Bhatia // Advances in Bioinformatics, Multimedia, and Electronics Circuits and Signals. – Springer, Singapore, 2020. – P. 155-168.
- [7] Арбузова, А.А. Диагностика легочных заболеваний с помощью нейронных сетей / А.А. Арбузова // Математическое и компьютерное моделирование естественно-научных и социальных проблем: материалы XIV

Международной научн.-техн. конф. молодых специалистов, аспирантов и студентов (Россия, г. Пенза, 14 июня 2020 г.) – 2020. – С. 185-189.

- [8] Cilibrasi, R. Clustering by compression / R. Cilibrasi, P. Vitanyi // IEEE Transactions on Information Theory. – 2005. – V. 51, Iss. 4. – P. 1523-1545.
- [9] Li, M. The Similarity Metric / M. Li, P. Vitányi // IEEE Transaction on Information Theory, vol. 50, N° 12, 2004, pp. 3250-3264.
- [10] Колмогоров, А.Н. Три подхода к определению понятия «количество информации» / А.Н. Колмогоров // Проблемы передачи информации. – 1965. – Т. 1, Вып. 1. – С. 3-11.
- [11] Верещагин, Н.К. Колмогоровская сложность и алгоритмическая случайность / Н.К. Верещагин, В.А. Успенский, А. Шень. – М.: МЦНМО, 2013. – 576 с.
- [12] Li, M. An Introduction to Kolmogorov Complexity and Its Applications. / Ming Li, P. Vitanyi – 3rd ed. New York: Springer-Verlag, 2008. – 809 p.
- [13] Quispe-Ayala, M.R. Image classification using data compression techniques / M.R. Quispe-Ayala, K. Asalde-Alvarez, A. Roman-Gonzalez // IEEE 26-th Convention of Electrical and Electronics Engineers in Israel. – 2010. – С. 000349-000353.
- [14] ChestX-ray14. [Электронный ресурс].
–URL: <https://nihcc.app.box.com/v/ChestXray-NIHCC>
(дата обращения 24.04.2021).
- [15] Chest X-Ray Images (Pneumonia)|Kaggle. [Электронный ресурс]. – URL: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia> (дата обращения 24.04.2021).
- [16] Shijie, J. Research on data augmentation for image classification based on convolution neural networks / J. Shijie, W. Ping, J. Peiyi and H. Siping // 2017 Chinese Automation Congress (CAC). – 2017. – P. 4165-4170.

- [17] Печников А.А., Прусский Д.А. Программы-архиваторы для вычисления колмогоровской сложности // Международный журнал прикладных и фундаментальных исследований, № 7. 2019. С. 118-123.
- [18] Старовойтов, В.В. Сравнительный анализ оценок качества бинарной классификации / В.В. Старовойтов, Ю.И. Голуб // Информатика. – 2020. – Т. 17. – №. 1. – С. 87-101.